Linearizing the Plenoptic Space

Grégoire Nieto, Frédéric Devernay, James L. Crowley

Linearizing the Plenoptic Space

Grégoire Nieto
Univ. Grenoble Alpes, Inria, LJK
Grenoble, France
gregoire.nieto@inria.fr

Frédéric Devernay
Univ. Grenoble Alpes, Inria, LJK
Grenoble, France
frederic.devernay@inria.fr

James Crowley
Univ. Grenoble Alpes, Inria, LIG
Grenoble, France
james.crowley@inria.fr

Abstract

The plenoptic function, also known as the light field or the lumigraph, contains the information about the radiance of all optical rays that go through all points in space in a scene. Since no camera can capture all this information, one of the main challenges in plenoptic imaging is light field reconstruction, which consists in interpolating the ray samples captured by the cameras to create a dense light field. Most existing methods perform this task by first attempting some kind of 3D reconstruction of the visible scene. Our method, in contrast, works by modeling the scene as a set of visual points, which describe how each point moves in the image when a camera moves. We compute visual point models of various degrees of complexity, and show that high-dimensional models are able to replicate complex optical effects such as reflection or refraction, and a model selection method can differentiate quasi-Lambertian from non-Lambertian areas in the scene.

1. Introduction

The light field has received much interest during the past decades, not only in the academic field, but also among consumers thanks to the availability of commercial plenoptic cameras such as *Lytro* and *Raytrix*. The concept was introduced from the study of the 5D plenoptic function [1], that returns the radiance along any ray going through any 3D point in the scene. If the region where the light field is measured contains no object, the radiance is the same for all points that lie on the same ray, so that the light field only has 4 dimensions. The optical device used to measure the light field (usually a plenoptic camera, or a set of standard cameras) samples the 4D ray space, just as a traditional camera samples the 2D space of rays going through a 3D single point. Reconstructing the light field consists in recovering the missing parts of the light field given the measured samples. In this work, we propose better local representation of the light field, which leads to a better reconstruction. The reconstructed light field can then be used to synthesize a novel view, as seen by a virtual camera that was not used to produce the initial ray samples.

Light field reconstruction is usually done by taking advantage of the epipolar constraint to estimate dense disparity maps [24]. This depth information is then jointly processed with source images to create a novel view [21, 17]. However, the epipolar constraint and the fact that a light ray may correspond to a given depth are based on strong assumptions made on the scene itself, which should be formed of solid shapes with an almost Lambertian reflectance. Real scenes may contain specular reflections, semi-transparent medium, refraction, or even inhomogeneous refractive index (as in mirages). If the original sampling of the 4D light field is dense enough, as happens in dense camera arrays or in so-called plenoptic cameras, the Lambertian assumption may be sufficient to locally describe the light field, and to approximate it in a small neighborhood of the original rays. However, as the light ray samples become more sparse, or if the novel view contains rays that lie outside of the 4D region of the ray space containing samples, any deviation of the real scene from the Lambertian assumption may be amplified and cause visual artifacts or non-realistic novel views.

We observe that deviations from the Lambertian assumption are mainly of two kinds. Some points belong to a solid shape but have a non-diffuse or anisotropic reflectance. These points belong to a surface, which may even be textured, and have a given depth, and the optical rays spanned by these points follow the rules of perspective projection and epipolar geometry: only their radiance deviates from

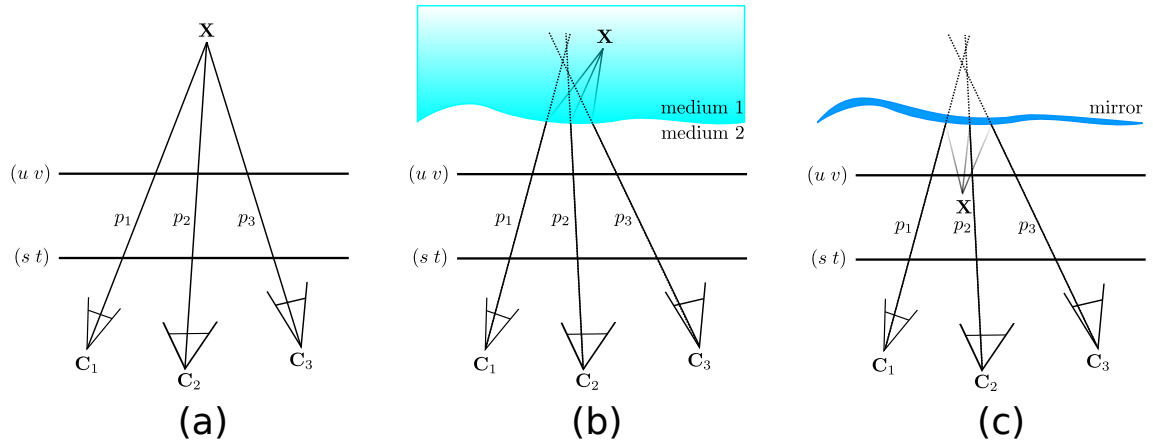


Figure 1. Geometric distortions of the light field. A visual point **X**, corresponding to a scene point, as seen by three cameras as three rays p_1, p_2 and p_3. (a) No distortion: all rays belong to a single pencil of lines and intersect at the 3D point. (b) Refraction by a change of optical medium bends the light rays. Rays do not necessarily intersect and triangulation fails to locate the point in space. (c) Likewise, mirror surfaces distort the rays, which may not intersect at a single 3D point.

the Lambertian assumption. Other non-Lambertian points in the images may correspond to complex optical paths, where the optical rays are affected by a series of reflections (or specular reflections), refractions (as when the rays switch between mediums of different refractive index, such as air, water and glass), or continuous variations of the refractive index. Although in some specific cases the reflection or refraction surfaces, and even the 3D points may be reconstructed [4], the general problem has too many unknowns since each optical ray may encounter many different material transitions. The common characteristic of this second kind of points is that they do not follow the common rules of perspective projection or parallax: when the eye or the camera moves to the left, an image point may move to the right (which is the expected behaviour), but may also move to the left, up or down, thus violating the epipolar constraints.

Based on these observations, we propose to focus on the reconstruction of the 4D light field itself, rather than on trying to explain the observed light field by reconstructing 3D surfaces and materials in the scene. When a 3D point that lies on a Lambertian surface is observed in the images, and if we suppose that there are no occlusions, the set of all optical rays that go through this 3D point, which form a pencil of lines, have the same radiance. The image of this point in any camera is given either by projecting the 3D point in the camera, or by taking the ray from the pencil of lines that goes through the camera optical center. Let us now suppose that the scene is static, and we observe a point in a camera that has a more complex behaviour due to reflections or refractions: its apparent motion in the image when the point of view changes slightly may not be consistent with parallax or epipolar geometry. We call it a *visual point*, and it consists of a two-dimensional set of rays, called a line congruence [18], which is more general than the pencil of lines: for a given camera optical center, the image(s) of this *visual point* corresponds to the element(s) of the line congruence that goes through the optical center. There is usually only one image, but there may be several, if there are several possible optical paths between the source and the optical center.

In this work, we propose to extract several rays corresponding to the same *visual point* by matching images (using optical flow), and then to fit simple linear line congruences to these visual points, corresponding to various levels of complexity. In our model, the radiance can also be a linear function of the ray parameters, thus modeling variations both in position and photometry. This can then be used to reconstruct missing rays in the light field, for example to compute all rays that go through a given camera optical center and render a novel view. To validate our approach, we perform experiments with sparse challenging light field datasets where we render novel views and compare with reference images. We also discriminate between several plenoptic models thanks to both qualitative and quantitative results.

Our main contributions are: a novel sampling technique and parametrization of the plenoptic space; an optimization process to fit complex models to the sampled plenoptic space, allowing more accurate reconstruction of specularities, transparencies and refractions; a new continuous rendering technique that satisfies most desirable properties an ideal image-based rendering (IBR) algorithm should have [5].

2. Related Work

Light Field Reconstruction The light field (also called the lumigraph) was initially used as an intermediate representation of the 4D radiance signal, which then could be processed to produce effects such as novel view synthesis, refocusing, matting, etc. [14, 9]. When sampling the 4D light field, there is a trade-off between angular and spatial resolution [6], which can be compensated by interpolation [8]. One way to interpolate the sparse samples in the 4D light field is to use a geometric proxy, which is a more or less precise reconstruction of the 3D scene, but computing a precise 3D reconstruction may turn out to be expensive [11, 12, 24], and this usually relies on the assumption that the scene is Lambertian, i.e. the radiance of a 3D surface point does not depend on the viewpoint. In this paper we do not explicitly reconstruct the scene geometry, since we only use pairwise ray matches between views, as given by optical flow, which may not correspond to the projections of an actual 3D point. Our goal is to handle generic scenes by using a more general model of the 4D plenoptic function [1].

Reconstructing reflective and specular scenes Interpolating the "flowed light field" as a way to render any viewpoint of the captured scene was experimented in [7]. However, they did not try to model the *visual points* of the scene, and only perform bilinear interpolation of the optical flow to synthesize a new view. In a computer graphics perspective, Zhou *et al*. [29, 28] model the non-Lambertian reflections by a Phong BRDF model. Tuning the Phong exponent allows them to model different types of shining surfaces (Lambertian, duller and specular) thus reducing the sampling rate of the light field required for novel view synthesis. Sulc *et al*. [23] separate the diffuse component from the specular component, which is estimated from the specular flow. It requires a precomputed disparity map based on the first order structure tensor [24]. Like other previously cited methods, it does not handle refractive surfaces.

Reconstructing refractive and transparent scenes In the context of light field, several papers already provide solutions to deal with transparent, translucent, or refractive surfaces, and sometimes even reconstruct them explicitly. Wetzstein *et al*. [25] suggest the use of a single camera with a lenslet array, and a lightbox is used in a way similar to photometric stereo, using varying colored light sources to encode spatial and angular domains. Iffa *et al*. [13] propose to split the optical flow into parallax flow and refractive flow (due to light deflection) with a single plenoptic camera shot. They solve a classic optical flow problem for every pair of computed flow at once, using a system of linear equations with a divergence-curl regularizer well known in the fluid flow tracking literature. A drawback is that it requires a highly textured background (which is not required by our method). Maeno *et al*. [15] introduced light field distortion features to describe and recognize an object composed of a refractive surface and a textured background, using a commercial plenoptic camera. Alterman *et al*. [3] use large displacement optical flow between two views to deal with refractions only. Like us, they propose a multi-view triangulation approach, although they only model Lambertian points seen through refractive media.

Summary Existing light field reconstruction methods have three main drawbacks. First, most methods are limited by the camera and scene setup (plenoptic cameras, light boxes, highly textured background). Apart from plenoptic cameras, other designs include camera arrays [26], which enabled the constitution of the Stanford Light Field Archive. Although we use these datasets in our experiments, our method can be applied to data captured by any plenoptic camera or by any set of cameras arranged in a generic configuration. Second, their goal is often an explicit representation of the scene geometry, which permits a better interpolation of the light field, but should not be necessary. In fact, an error on the scene reconstruction may have a dramatic impact on the light field reconstruction. Our method works directly on light field interpolation, without an explicit scene reconstruction. Third, only one issue is tackled at a time: either they try to separate the diffuse component from the specular one, or they deal with refractive surfaces. But both problems are never addressed simultaneously. Our method works by modeling any peculiar light behaviour, which we locally deal with by a better approximation of the light flow.

3. Overview

Our method is composed of several key steps. We first compute the optical flow between pairs of adjacent views to create a set of color and position samples attached to a single *visual point*, which we call the *light flow*. Each sample in this set is a ray defined by the 4 parameters giving its orientation in space (section 4) and its radiance. When the capture device is a set of cameras, each ray passes through the optical center of the camera that sees it, and the radiance is taken from its pixel color value.

Then, we fit a line congruence model to each sample set (section 5), which aims to explain the motion of the visual point in the 4D light field parametrization. The 3-parameter model corresponds to all rays passing through a single 3D point, and we also devise linear congruence models with 4 or 6 parameters, in order to predict phenomena such as refraction, reflection, or optical index variations. A model selection method handles the trade-off between the com-

plexity of the model (in term of number of parameters) and the error residual, in order to avoid data over-fitting.

Finally we synthesize a novel view by interpolating the projection of the visual points in the new camera sensor plane (section 6). For each visual point, its position in the target image is found by intersecting the line congruence model with the optical center of the target camera. The scene is rendered by accumulating colors of every *visual point*, a technique known as splatting in the computer graphics literature. The color to splat is estimated by fitting a linear model to the color samples of the visual point.

4. Plenoptic Sampling and Parametrization

The plenoptic space is the space of light rays that pass through a scene, through any point and in any direction. We assume that in the region of space where we want to reconstruct the light field, the radiance along a ray is constant. As in [14], we use the 4D light field parametrization of rays called *light slab*, where the coordinates (u, v, s, t) are obtained by intersecting the 3D ray with two parallel 3D reference planes, where (u, v) and (s, t) are the coordinates of these intersections within each plane. Let us note I the radiance of this ray, represented by 3 more coordinates, corresponding to its RGB color components. Each ray is thus described by 7 coordinates.

Let us consider a 3D point in space. The set of rays that emanate from this this point is a 2-dimensional set of lines, called a line congruence. Since all lines go through the same point, this congruence is reduced to a 2D pencil of lines.

In most cases, the optical paths between this 3D point and the region where we want to reconstruct the light field is free of occluders and is of homogeneous optical index. In this case, the pencil of lines is not modified by the optical medium (figure 1a). However, if there are refractions, reflections, or variations of the optical index, the pencil of lines is distorted into a more generic line congruence (figures 1b, 1c). Such a line congruence, noted P, describes the 2D set of rays in the plenoptic space that corresponds to what we call a *visual point*. The 7-dimensional coordinates of rays that belong to the same line congruence are strongly correlated. For instance rays corresponding to a Lambertian 3D point seen through free space (figure 1a) have a constant radiance, and coordinates (u, v, s, t) span a plane in 4D parametrized by the Cartesian coordinates of the 3D point. The line congruence is fully represented by a point in a 3D space and its radiance (i.e. 3+3 parameters). More complex line congruence models described by a higher number of parameters may describe more complicated optics [19], and in this paper we restrict ourselves to linear models presented in section 5, which can be used to describe faithfully the local geometry of these line congruences.

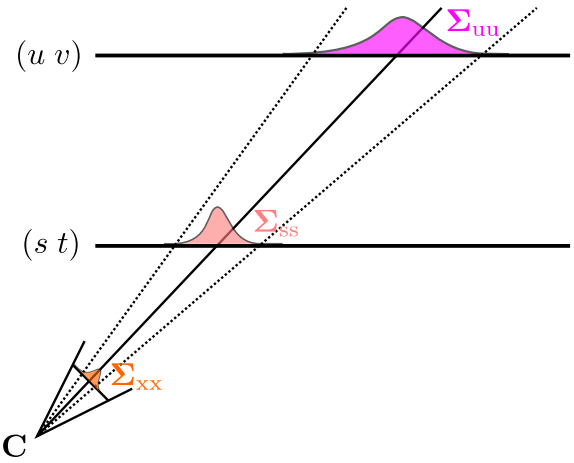


Figure 2. Propagation of geometric uncertainty from the image plane of the camera to the 2-plane parametrization. $\mathbf{\Sigma_{xx}}$ is the original matching uncertainty in the image plane. $\mathbf{\Sigma_{ss}}$ and $\mathbf{\Sigma_{uu}}$ are the variances of the marginal distributions of $\mathbf{s}$ and $\mathbf{u}$ respectively.

Sampling A camera image contains sample rays that go through the image plane and the optical center of the camera. Usually, in the line congruence that corresponds to a visual point, only one ray meets the optical center of a camera, so that several cameras are required to get several sample rays from the same visual point.

Let us consider a ray in a reference image. The corresponding rays in the other images can be obtained from the optical flow, but any other point matching method that is not constrained by the epipolar geometry can be used (non-Lambertian points do not respect the epipolar constraint). The optical flow is only computed between neighbors in the camera setup (see Figure 7), so that we can assume that the appearance between the two views is not too different. A *visual point* is thus represented by the list of its position and radiance in images where it is visible. Considering that optical flow is dense, we have as many vectors or sample set as pixels in source images. Each image point is then converted to a 4D light slab as explained below.

Ray parametrization Given a source camera described by its optical center $\mathbf{C}$, its rotation $\mathbf{R}$, and its matrix of intrinsic parameters $\mathbf{K}$, the 4D light slab representation of an image point $\mathbf{x}$ is obtained by intersecting the 3D ray going through $\mathbf{x}$ and $\mathbf{C} = (C_x, C_y, C_z)$ with the two parallel planes. Without loss of generality, we can assume the two planes are of equation $z = 0$ and $z = 1$. Let $\mathbf{s} = (s, t)$ be the intersection with the plane of equation $z = 0$ and $\mathbf{u} = (u, v)$ be the intersection with the plane of equation $z = 1$. The direction vector of the ray $\mathbf{r} = (r_x, r_y, r_z)$ can be obtained as

$$\mathbf{r} = \mathbf{R}^\intercal \mathbf{K}^{-1} \begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix}, \qquad (1)$$

and the light slab coordinates are

$$s = C_x - C_z\frac{r_x}{r_z}, \qquad t = C_y - C_z\frac{r_y}{r_z}, \tag{2}$$

$$u = C_x + (1 - C_z).\frac{r_x}{r_z}, \quad v = C_y + (1 - C_z).\frac{r_y}{r_z}. \tag{3}$$

Uncertainty of measurements In the classic triangulation problem, the 3D point model is fitted to the data (image points) by minimising the reprojection error. The errors are usually considered to be Gaussian and isotropic in each image plane with an identical variance. In our case, we need to fit a line congruence model to a set of rays parametrized by (s, t, u, v), given the matching error, which is also measured in the images. Therefore we need to express the covariances of the intersections with the two planes (s, t) and (u, v). This covariance is derived by propagating the uncertainty from the image point to the planes. The covariance of (u, v, s, t) is noted $\mathbf{\Sigma_{u,s}}$. The covariance matrices of the marginal errors on $\mathbf{s}$ and $\mathbf{u}$ are noted $\mathbf{\Sigma_{ss}}$ and $\mathbf{\Sigma_{uu}}$ respectively. The Jacobian matrices of the parametrization are

$$\frac{\partial \mathbf{u}}{\partial \mathbf{x}} = (1 - C_z).\mathbf{J_r}, \qquad \frac{\partial \mathbf{s}}{\partial \mathbf{x}} = -C_z\mathbf{J_r}, \tag{4}$$

with $\mathbf{J_r} = \frac{\partial \mathbf{r}}{\partial \mathbf{x}} = \frac{1}{\mathbf{r}_z}(\mathbf{I}_2| - \mathbf{r})\mathbf{R}^\intercal\mathbf{K}^{-1}(\mathbf{I}_2|\mathbf{0}_2)^\intercal$. We note $\mathbf{S} = \mathbf{J_r}\mathbf{\Sigma_{xx}}\mathbf{J_r}^\intercal$ the uncertainty on the direction vector of the ray. The covariance matrix $\mathbf{\Sigma_{xx}}$ represents the matching uncertainty of the optical flow. Although we used a fixed point matching uncertainty in our experiments, most optical flow method also output a quality image that could be used to modulate this uncertainty (which may be for example larger in texture-less areas).

It should be observed that the matching uncertainty grows when a match between two rays is computed from chained optical flows. Assuming that this error is normally distributed, the covariance matrices of the flows add up, and the covariance of the chained flow is proportional to the number of flows: $\mathbf{\Sigma_{xx}}$ for direct matches, $2\mathbf{\Sigma_{xx}}$ for matches computed using a chain of two optical flow matches, etc. In our setup, since we compute optical flow from neighboring views only, views that are distant from the reference view have a larger matching error, and contribute less to the model than closer views (Figure 7).

Finally, we obtain the covariance matrices of the marginal errors

$$\mathbf{\Sigma_{uu}} = (C_z - 1)^2\mathbf{S}, \tag{5}$$

$$\mathbf{\Sigma_{us}} = \mathbf{\Sigma_{su}} = (C_z - 1)C_z\mathbf{S}, \tag{6}$$

$$\mathbf{\Sigma_{ss}} = C_z^2\mathbf{S} \tag{7}$$

and the covariance matrix of the joint distribution.

$$\mathbf{\Sigma_{u,s}} = \begin{pmatrix} \mathbf{\Sigma_{uu}} & \mathbf{\Sigma_{us}} \\ \mathbf{\Sigma_{su}} & \mathbf{\Sigma_{ss}} \end{pmatrix}. \tag{8}$$

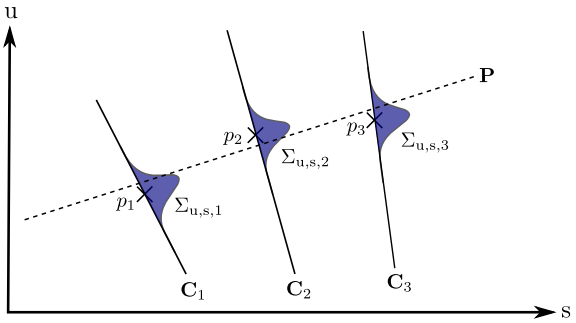


Figure 3. A linear geometric model of the pencil of rays $\mathbf{P}$ is fitted to the data. Sample rays of three views are noted p_1, p_2 and p_3. The associated covariances of the joint geometric distribution are noted $\mathbf{\Sigma_{u,s},1}$, $\mathbf{\Sigma}_{u,s,2}$ and $\mathbf{\Sigma_{u,s}},3$. They weight the contribution of each view in the optimization process.

$\mathbf{\Sigma_{u,s}}$ is a real symmetric matrix of rank 2, and the associated 2-dimensional linear subspace spans the set of rays that go through the optical center of the source camera. Thus, $\mathbf{\Sigma_{u,s}}$ represents an error on a point that lies on a 2D plane in the 4D light field space.

The uncertainty of the radiance measurement is derived from the matching uncertainty of the optical flow. The covariance matrix of the radiance error has the expression

$$\mathbf{\Sigma_{I,s}} = \begin{pmatrix} \mathbf{\Sigma_{II}} & \mathbf{\Sigma_{Is}} \\ \mathbf{\Sigma_{sI}} & \mathbf{\Sigma_{ss}} \end{pmatrix}, \tag{9}$$

where the covariance matrices of the marginal errors are

$$\mathbf{\Sigma_{II}} = \nabla\mathbf{I}\mathbf{\Sigma_{xx}}\nabla\mathbf{I}^\intercal \tag{10}$$

$$\mathbf{\Sigma_{Is}} = -C_z\nabla\mathbf{I}\mathbf{\Sigma_{xx}}\mathbf{J_r}^\intercal \tag{11}$$

$$\mathbf{\Sigma_{sI}} = -C_z\mathbf{J_r}\mathbf{\Sigma_{xx}}\nabla\mathbf{I}^\intercal \tag{12}$$

$$\mathbf{\Sigma_{ss}} = C_z^2\mathbf{J_r}\mathbf{\Sigma_{xx}}\mathbf{J_r}^\intercal. \tag{13}$$

5. Plenoptic Space Modeling

Given the 4D data we obtain thanks to the optical flow and the parametrization previously detailed, we are able to fit a geometric model. We first detail the simple model of the line congruence corresponding to a 3D point $\mathbf{X}$, which is a pencil of lines. Then we derive linear geometric models of line congruences with more than 3 parameters.

Geometric model Let P be the pencil of lines that pass through $\mathbf{X} = (x, y, z)$. For every line $\mathbf{q} = (u, v, s, t)$ passing through $\mathbf{X}$ we have

$$\mathbf{q} \in P \iff \begin{cases} u = \alpha s + \beta_u \\ v = \alpha t + \beta_v \end{cases}, \tag{14}$$

with

$$\alpha = \frac{z - 1}{z}, \ \beta_u = \frac{x}{z} \text{ and } \beta_v = \frac{y}{z}. \tag{15}$$

We find an estimation of $\mathbf{X}$ by solving the linear system above for α, β_u and β_v, which is equivalent to the classic triangulation of a point. The number of parameters defines the dimensionality of the *visual point*. We name this model 3g, refering to the 3 geometric parameters that define it.

We can easily extend this model to line congruences that follow a linear equation, as in eq. (14), but may not pass through a point in the 3D space. In a more generic way we can write:

$$\mathbf{q} \in P \iff \mathbf{u} = \mathbf{A}\mathbf{s} + \mathbf{b}. \tag{16}$$

In the previous case where all rays intersect in a 3D point in space, $\mathbf{A} = \alpha \mathbf{I}_{22}$ and $\mathbf{b} = (\beta_u, \beta_v)$. We introduce two other models, 4g:

$$\mathbf{A} = \begin{pmatrix} \alpha_u & 0 \\ 0 & \alpha_v \end{pmatrix}, \ \mathbf{b} = \begin{pmatrix} \beta_u \\ \beta_v \end{pmatrix}, \tag{17}$$

and 6g:

$$\mathbf{A} = \begin{pmatrix} \alpha_{us} & \alpha_{ut} \\ \alpha_{vs} & \alpha_{vt} \end{pmatrix}, \ \mathbf{b} = \begin{pmatrix} \beta_u \\ \beta_v \end{pmatrix}. \tag{18}$$

Computing the 3, 4 or 6 geometric parameters of the line congruence can be posed as a least square problem. We are given a set of K 4D samples $\{\mathbf{p}_1, \ldots, \mathbf{p}_K\}$ and we try to fit a model (with 3, 4 or 6 parameters) by minimizing the sum of squared Mahalanobis distances to the data samples, given the rank-2 covariance of the measurement error on each data sample $\mathbf{\Sigma}_{\mathrm{u,s}}$. It is possible to define a Mahalanobis distance with this matrix only if the residual vector lies in the subspace spanned by the eigenvectors associated to the two non-null eigenvalues μ_1 and μ_2. This assumption provides additional constrains on the construction of the residual.

Expression of the residual Let $\mathbf{p} = (p_u, p_v, p_s, p_t)$ be a sample ray, corresponding to a point in a camera, and let $\mathbf{r} = (r_u, r_v, r_s, r_t)$ be its residual error: $\mathbf{q} = \mathbf{p} + \mathbf{r} \in P$. We recall that P is the line congruence associated with the *visual point*. $\mathbf{r}$ is constrained to lie in the subspace spanned by the two eigenvectors $\mathbf{e_1}$ and $\mathbf{e_2}$, associated respectively with μ_1 and μ_2. Our goal is to find the parameters of the model that minimize the squared Mahalanobis norm of $\mathbf{r}$. On one hand we have

$$\mathbf{p} + \mathbf{r} \in P \iff \begin{pmatrix} p_u + r_u \\ p_v + r_v \end{pmatrix} = \mathbf{A} \begin{pmatrix} p_s + r_s \\ p_t + r_t \end{pmatrix} + \mathbf{b} \tag{19}$$

and on the other hand

$$\mathbf{r} = r_1 \mathbf{e}_1 + r_2 \mathbf{e}_2. \tag{20}$$

By substitution, we obtain the expression of the residual in a basis of eigenvectors:

$$\begin{pmatrix} r_1 \\ r_2 \end{pmatrix} = (\mathbf{E_u} - \mathbf{A}\mathbf{E_s})^{-1} \left(\mathbf{A} \begin{pmatrix} p_s \\ p_t \end{pmatrix} + \mathbf{b} \right), \tag{21}$$

with

$$\mathbf{E_u} = \begin{pmatrix} \mathbf{e}_{1,u} & \mathbf{e}_{2,u} \\ \mathbf{e}_{1,v} & \mathbf{e}_{2,v} \end{pmatrix} \text{ and } \mathbf{E_s} = \begin{pmatrix} \mathbf{e}_{1,s} & \mathbf{e}_{2,s} \\ \mathbf{e}_{1,t} & \mathbf{e}_{2,t} \end{pmatrix}. \tag{22}$$

The cost function For each sample $\mathbf{p}_k, k \in [1, K]$ in the sample set, let $\mathbf{r}_k$ be its residual and $\mu_{k,1}$ and $\mu_{k,2}$ the eigenvalues associated with the sample $\mathbf{p}_k$. The cost function, is

$$\|f(\mathbf{A}, \mathbf{b})\|^2 = \sum_{k=1}^{K} \|f_k(\mathbf{A}, \mathbf{b})\|^2_{\mathbf{D_k}}, \tag{23}$$

where

$$\|f_k(\mathbf{A}, \mathbf{b})\|^2_{\mathbf{D_k}} = \mathbf{r}_k^\intercal \mathbf{D}_k^{-1} \mathbf{r}_k, \text{ with } \mathbf{D}_k = \begin{pmatrix} \mu_{k,1} & 0 \\ 0 & \mu_{k,2} \end{pmatrix}. \tag{24}$$

In the case of the 3g model, it can be shown that this optimization is equivalent to a classic triangulation with bundle adjustment. But instead of minimizing the sum of the squared reprojection error, we minimize the sum of squared errors of the rays. Each contribution is weighted by the inverse of the uncertainty propagated from the image plane to the line slab parametrization. This formalism is very valuable because it allows the modelling of complex *visual points*, with more than 3 parameters.

Photometric model Assuming that the visual point is Lambertian, all the rays of the visual point P have the same color whatever their direction. It means that its radiance $\mathbf{I} = (R, G, B)$ is constant with respect to $\mathbf{s}$ and that the photometric model has thus 3 parameters. Let us name this model 3p.

In the general case, the Lambertian assumption is not necessarily verified: the radiance of the visual point depends on the point of view. Similarly to the geometry of the plenoptic space, we linearize the color $\mathbf{I}$ as function of the angular displacement: $\mathbf{I}(\mathbf{s}) = \mathbf{A}\mathbf{s} + \mathbf{I}_0$. The number of parameters to find is 9 (6 for the $\mathbf{A}$ matrix, and 3 for $\mathbf{I}_0$). There are $3K$ scalar measurements (3 channels times the number of views that see the point). Let us name this model 9p.

Each color sample is weighted by the inverse of the joint distribution variance, $\mathbf{\Sigma_{I,s}}$, and the parameters are solved by least squares.

Model selection Fitting a model with more parameters generally leads to a lower fitting error, but when the number of parameters to estimate gets close to the number of samples, there is a risk of overfitting the measurement, resulting in a wrong interpolated motion of the visual point. Model selection techniques are commonly used to discriminate between different models varying in fitting performance and

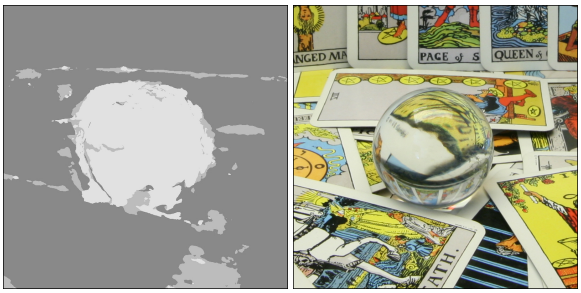


Figure 4. Geometric model selection by BIC on tarot coarse dataset. Light grey: $6\mathrm{g}+9\mathrm{p}$. Middle grey: $4\mathrm{g}+9\mathrm{p}$. Dark grey: $3\mathrm{g}+9\mathrm{p}$. BIC discriminates between Lambertian (tarot cards) and refractive of specular areas (glass ball).

number of parameters. After having estimated the parameters of our three geometric models, we apply a Bayesian information criterion (BIC) [22] to select the best model for each *visual point*. Denoting $\hat{\mathbf{A}}$ and $\hat{\mathbf{b}}$ the matrix and the vector that contain the estimated parameters, the formula of the BIC is

$$\mathrm{BIC} = \|f(\hat{\mathbf{A}}, \hat{\mathbf{b}})\|^2 + n.\ln K \tag{25}$$

where K is the number of samples, n the number of parameters and $\|f(\hat{\mathbf{A}}, \hat{\mathbf{b}})\|_2^2$ is -2 times the log-likelihood of the estimated parameters. For each sample set we select the model that minimizes the BIC. Figure 4 shows the result of model selection performed on the tarot dataset from the Stanford Light Field Archive. One can notice that the $3\mathrm{g}+9\mathrm{p}$ model is sufficient for most diffuse and opaque areas such as the tarot cards, but is supplanted by the more complex geometric models such as $6\mathrm{g}+9\mathrm{p}$ in refractive areas like on the transparent ball.

6. Rendering

We demonstrate a sample application of our *visual point* models by performing novel view synthesis. Given a target view defined by known camera parameters $\mathbf{C}$, $\mathbf{R}$ and $\mathbf{K}$, and for each visual point of the scene, our goal is to find the position and radiance of the line from the *visual point* that passes through the optical center $\mathbf{C}$. The goal of novel view synthesis is to find the color of each pixel in the novel view. To this end, we should find, for each pixel in the target view, which *visual points* have a line inside this pixel, and then mix their radiances (this is usually called *backward warping*). In practice, this problem would require a lot of computation, and we thus prefer *forward warping* every reconstructed *visual point* by computing the line of this visual point that passes through the camera optical center, and by painting the pixels accordingly.

For each *visual point* P, modeled by a linear line congruence, we find the corresponding light ray captured by the target view as the intersection of the line congruence

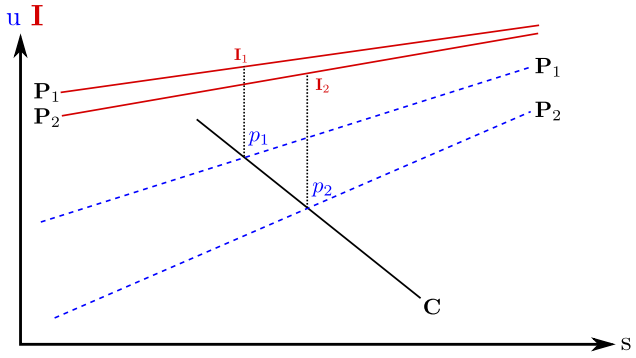


Figure 5. Light flow rendering. The black line is the set of rays that pass through the target camera optical center $\mathbf{C}$, $\mathbf{P}_1$ and $\mathbf{P}_2$ are two *visual points* models. The blue lines represent the linearized geometric relationship between $\mathbf{u}$ and $\mathbf{s}$, the red lines represent the photometric relationship between $\mathbf{I}$ and $\mathbf{s}$. Interpolated rays (which correspond to points in the camera) are the intersections p_1 and p_2, and their interpolated radiances are $\mathbf{I}_1$ and $\mathbf{I}_2$.

(which is a plane in the 4D light field) with the pencil of lines that corresponds to the camera (see figure 5). For example the intersection between the $6\mathrm{g}$ *visual point* represented by $(\alpha_{u,}, \alpha_{ut}, \alpha_{vs}, \alpha_{vt}, \beta_{cu}, \beta_{cv})$ and the camera with optical center $\mathbf{C} = (C_x, C_y, C_z)$ has $\mathbf{s}$-coordinates

$$\mathbf{s} = \begin{pmatrix} \alpha_{us} - \alpha_c & \alpha_{ut} \\ \alpha_{vs} & \alpha_{vt} - \alpha_c \end{pmatrix}^{-1} \begin{pmatrix} \beta_{cu} - \beta_u \\ \beta_{cv} - \beta_v \end{pmatrix} \tag{26}$$

with

$$\alpha_c = \frac{C_z - 1}{C_z},\ \beta_{cu} = \frac{C_x}{C_z},\ \beta_{cv} = \frac{C_y}{C_z}. \tag{27}$$

The $\mathbf{u}$-coordinates can then be computed from either the linear *visual point* model or the camera optical center. It is now easy to derive the associated image point, by projecting the point $(s\ t)$ on the camera sensor plane:

$$\mathbf{x} = \mathbf{KR}\left(\begin{pmatrix} s \\ t \\ 0 \end{pmatrix} - \mathbf{C}\right). \tag{28}$$

Likewise we find the color of the ray by substituting of the $\mathbf{s}$-coordinates in the estimated photometric model (see figure 5).

Note that some of these points can fall outside of the field of view of the target, since not all *visual points* may be visible in a given view. Once the destination image point is computed, we splat the color of the visual point onto the surrounding pixels. Splatting is a well-known technique of point-based rendering [10] that consists in accumulating rendered 3D flat primitives (usually disks or ellipsoids) centered on the 3D point and oriented by a reconstructed normal. Because we do not have any information about the normals and rendering elliptical splats usually blurs the image, we instead accumulate uniform squared splats with bilinear contributions to neighboring pixels.

Splatting is normally done in three passes: visibility computation, blending and normalization. We skip the first one since computing a z-buffer only makes sense when depth is defined. Only the 3g model contains information about depth ($z = 1/(1-\alpha)$), but even this perceived depth may be an illusion (as in the "floating coin illusion" which uses a parabolic mirror). Thus we blend all the visual point projections in the target view, accumulating RGB color and contributions in an alpha channel. Finally, we normalize the image by dividing by the alpha channel.

Epipole consistency As mentioned in [5], a ray that passes through the center of projection of a source camera "should be trivially reconstructed from the ray database". Our rendering algorithm does not exactly fulfill this property since the rays forming the *visual point* do not necessarily belong to the original sample set, *i.e*. the model does not exactly fit the data. It would be the case if the fitted model passed through all the 4D sample points. However, the optimization makes sure that the estimated model is as close as possible to the sample set by minimizing the Mahalanobis distance.

Minimal angular deviation An IBR algorithm should ensure that input views that are close to the target view should contribute more to the rendered color. Angular deviation is a usual measure of "closeness". Such a property is fulfilled by our algorithm thanks to the fact that we model the radiance as a linear function of the angular coordinates (s, t). Since a ray that is close to another in angle is also close with respect to the (s, t) coordinates, its radiance should be similar.

Resolution sensitivity The color of the visual point can be interpolated from the input color samples as most IBR direct approaches do: a weighted average of the source images. Each weight depends on the capability of the source view to gather information about the scene. In other terms, a source view contributes more to the final color if it is close to the observed *visual point*, or has a high focal length. This role is played by the covariance $\mathbf{\Sigma_{I,s}}$ that weights the source views when fitting the photometric model. The farther the camera is, or the lower its resolution, the bigger the uncertainty will be. It can be seen as a cone of uncertainty projected on the plane $z = 0$. As a consequence, a ray with a large covariance matrix will contribute less to the model fitting. Conversely, a camera that is close to the scene or has a long focal length measures precisely the *visual point* which leads to a small covariance.

Continuity This rendering method assures the continuity with respect to the change of viewpoint. When we move continuously the target viewpoint, the camera plane moves continuously in the 4D space, and since the position and radiance of visual points is computed by intersecting planes in 4D, the position and radiance also move continuously, which is also a "desirable" property of IBR [5].

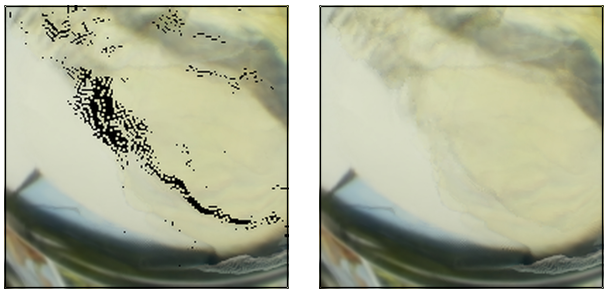


Figure 6. `PUSH/PULL` hole filling. Holes are caused by optical flow dilatation, self-occlusion or very far extrapolation. A very distant view has been synthesized in this example to highlight these holes and how the inpainting algorithm fills them.

Hole Filling Due to the use of forward warping, areas in the final image may end up with no radiance information, resulting in holes (see figure 6). Any hole filling or inpainting algorithm which is continuous with respect to the input image may be used to solve this problem. We used the push-pull inpainting algorithm as it was introduced in [9]. It is equivalent to performing isotropic diffusion in the areas to fill.

7. Experiments

Our method aims to be as generic as possible, and is thus not limited to a specific camera type or configuration. We did our experiments on images from a dataset captured with a camera array [27], and used the original non-rectified images from this dataset.

We started with a multiview camera calibration made using openMVG [16], which computes camera extrinsic and intrinsic parameters and removes camera distortion (which is not taken into account in our model). The placement of cameras we use in our experiment, described on the figure 7, is composed of 24 views arranged in an 3×3 internal square and a 5×5 external square of cameras – we removed the central view. It was originally a 9×9 array from which we removed every second row and column to exacerbate the sparsity of the light field. Removed views are kept aside for comparison and numerical evaluation. An open source optical flow algorithm [20] is run over each pair of neighboring views, so that we get as many samples as computed flows (see figure 7). The strategy to fill the sets of samples is the following:

- we compute the flow from the central view to the views in the internal square,

- we append the positions in the internal views to the respective sample sets,
- the optical flow is computed from the internal views to the neighboring external views,
- starting from each previous position, new sample positions are found via the computed flows.

Once the sample ray sets are filled with colors and 2D positions, converted to 4D via the light field parametrization (see section 4), we compute the visual point models using the Ceres Solver [2], which minimizes the L^2-norm of the block residual $f_k(\mathbf{A}, \mathbf{b}) = \left(\frac{r_{k,1}}{\sqrt{\mu_{k,1}}}, \frac{r_{k2}}{\sqrt{\mu_{k2}}}\right)$ with the algorithm `DENSE_QR`. Four models are tested to fit the 4D light field samples: $3\mathrm{g} + 3\mathrm{p}$, $3\mathrm{g} + 9\mathrm{p}$, $4\mathrm{g} + 9\mathrm{p}$ and $6\mathrm{g} + 9\mathrm{p}$. The first number accounts for the number of parameters of the linear line congruence model ($\mathbf{u}$ as a function of $\mathbf{s}$), while the second one accounts for the number of parameters of the photometric model ($\mathbf{I}$ as a function of $\mathbf{s}$). A $3\mathrm{g} + 3\mathrm{p}$ model would correspond to a 3D point with a constant RGB color, which models a Lambertian point. Using our algorithm, we render the central view, a view in the top left corner between the internal and the external square, and we extrapolate a view to the left, outside of the sampled light field region. Resulting views are cropped to 800×800 images.

Figure 8 shows the resulting central view, aside with the original image we removed, the absolute difference and the final residual of 3-parameter model optimization. Apart from tarot coarse, all results are very close to the original images. Most artifacts occur in the glass ball, where light rays are bended by refraction. The 3-parameter fails to model the behaviour of the distorted light since rays that emanate from the same point in the background (diffuse cards) are unlikely to intersect. Models with 4 and 6 parameters produce better results, as can attest the figure 9. The same interpretation can be claimed for specularities on the treasure chest or on the bracelet. In addition the bracelet dataset shows that in non-textured regions the model fails to fit the data samples altered by inaccurate optical flow. Nevertheless it does not affect the rendering because the interpolated position of the splat is frivolous in low-textured regions of the image. The figure 9 shows close-ups of synthesized views of the tarot dataset, to demonstrate the effect of rendering with different models. The more parameters we use, the more faithful to the original images the results are. This is supported by numerical results in table 1. Few differences between models are visible when rendering the central view, but extrapolation clearly discriminates the consequence of rising the dimensionality of the searched *visual point*.

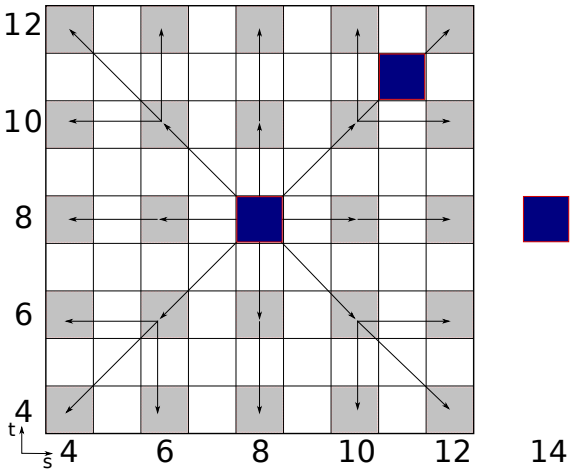


Figure 7. Light field setup for experiments. Cameras from the Stanford dataset are arranged on the same plane (s, t). Each cell is a view from the original dateset. Grey cells are views that are used to sample the plenoptic space. Arrows indicate how the optical flow is performed. Blue cells are view we synthesize to evaluate our method. We interpolate views $(8, 8)$ and $(11, 11)$, and extrapolate view $(14, 8)$.

	View $(8,8)$		View $(11,11)$		View $(14,8)$	
	PSNR	DSSIM	PSNR	DSSIM	PSNR	DSSIM
$3\mathrm{g} + 3\mathrm{p}$	26.37	64	23.61	109	24.00	102
$3\mathrm{g} + 9\mathrm{p}$	26.34	64	23.65	109	24.85	99
$4\mathrm{g} + 9\mathrm{p}$	26.32	64	25.08	89	24.71	96
$6\mathrm{g} + 9\mathrm{p}$	**26.44**	**63**	**25.57**	**74**	**27.20**	**69**

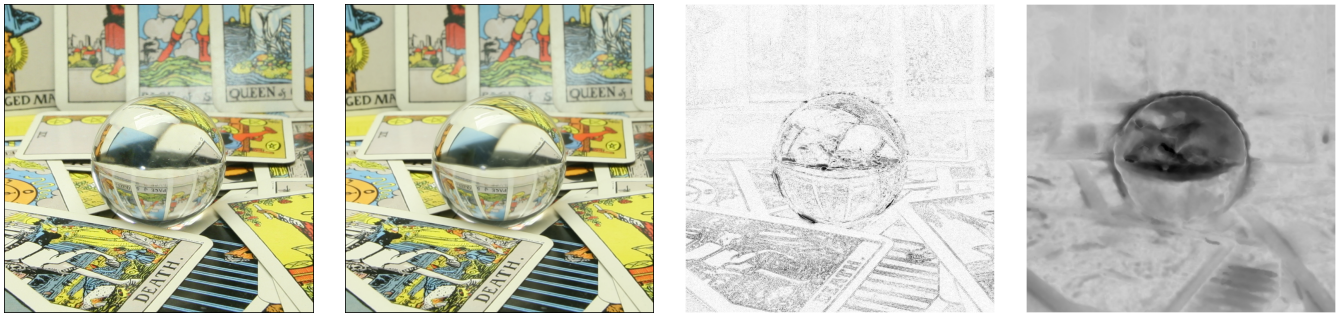
Table 1. Numerical results on tarot coarse dataset.

8. Conclusion

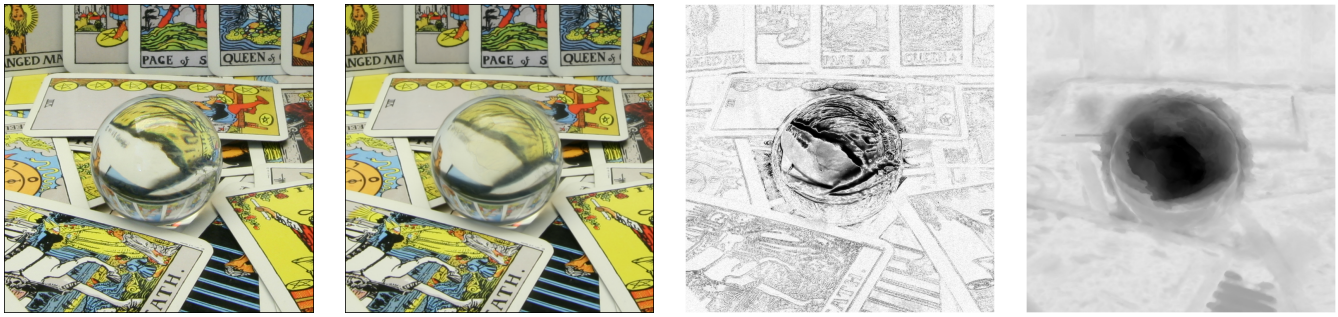
This paper presented a novel approach to light field reconstruction, based on a linear approximation of the line congruences that form the 4D light field. Whereas most light field reconstruction methods compute first a 3D representation of the scene, our method works directly on how the scene is *perceived* through the images, without attempting an explicit 3D reconstruction.

In this representation, each *visual point* is represented by geometric and photometric parameters. The geometric representation of a visual point is a 2D set of lines, also called a line congruence, which contains information on how the point moves in the image when the camera moves. The photometric parameters contain information on how the radiance of this point varies as a function of the viewpoint. For example, a point on a Lambertian surface is represented by a pencil of lines going through the 3D point, and a single color, which makes 3 geometric and 3 photometric parameters.

We devise models with 3, 4, or 6 geometric parameters, and 3 or 9 photometric parameters, which can model optical effects such as reflections, refraction, or variation in the



Tarot fine PSNR: 35.03 DSSIM: 16



Tarot coarse PSNR: 26.34 DSSIM: 64

Bracelet PSNR: 39.93 DSSIM: 5

Chest PSNR: 35.01 DSSIM: 272

original image result absolute difference final cost

Figure 8. Results on several datasets from the Stanford Light Field Archive. We compare the synthesized result with the original image. We also display the absolute difference between the two and the final value of the cost function. The model $3g + 9p$ is used in this experiment. Notice that the highest error values are located on refractive and specular areas. In the *bracelet* results, the black area in the final cost image denotes a wrong reconstruction which does not affect the result (low absolute difference error) because of the lack of texture in this area.

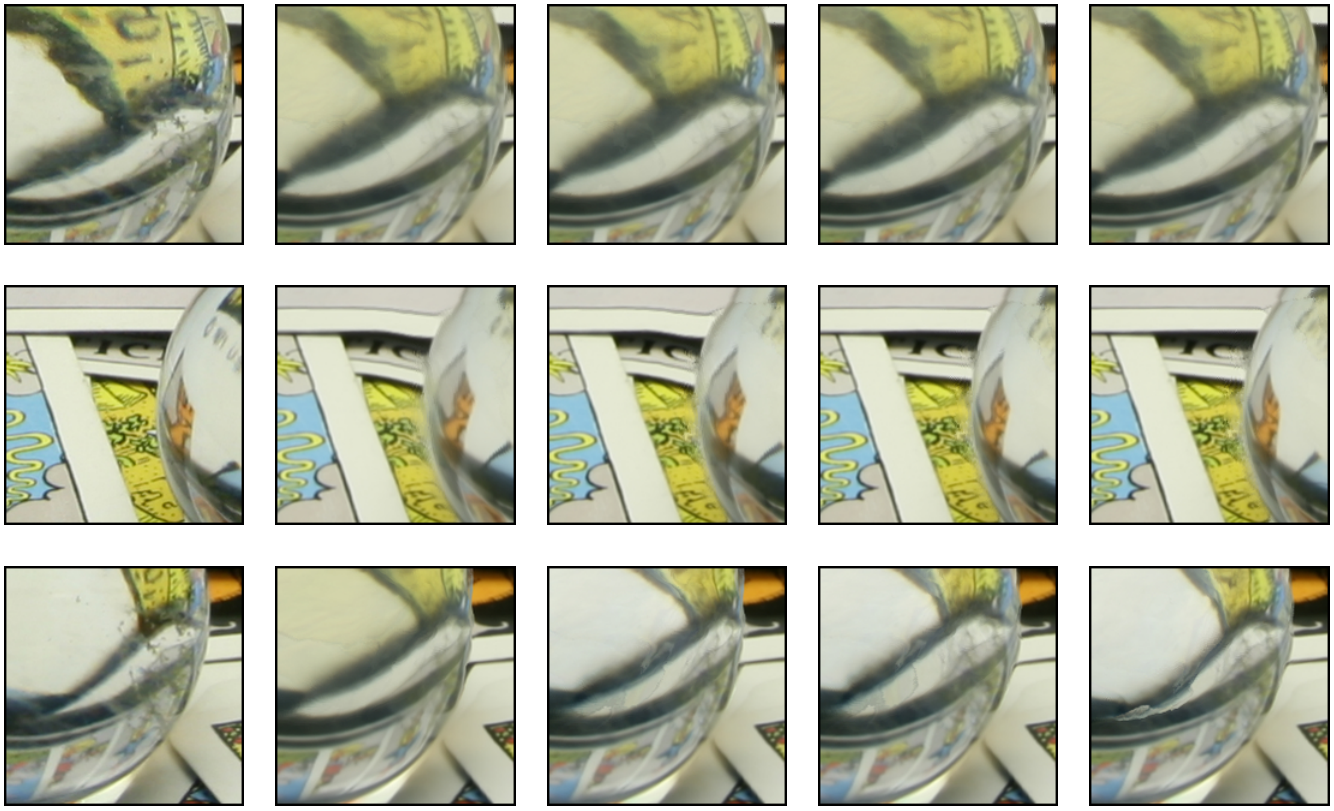


Figure 9. Results on challenging parts of the tarot coarse dataset. Top row: center view $(8, 8)$. Middle row: top-right view $(11, 11)$. Bottom row: extrapolated view $(14, 8)$. It exhibits main artifacts that are partially fixed by visual point models with more parameters.

optical index, as well as non-Lambertian surfaces.

Experiments show that the various *visual point* models are able to cope with complex optical phenomena that cannot be modeled by a 3D reconstruction. A model selection method is able to separate points in the scene that are Lambertian or quasi-Lambertian from *visual points* that cannot be modeled by a simple pencil of lines.

The *visual point* model could be further enriched, for example by modeling nonlinear line congruences, as those that can be caused by spherical or cylindrical surfaces. The rendering algorithm could also take into account the visibility of each visual point, and render a given visual point only if the rays that were used to compute the model are close enough to the synthesized viewpoint. Another extension would be to incorporate the time dimension in our visual point models, and compute time-varying line congruences, which could be used for novel view synthesis from asynchronous video sequences or from a set of photographs taken at different times and from different places.

Acknowledgment We thank the French Direction Générale de l'Armement for funding this work.

References

[1] E. H. Adelson and J. R. Bergen. The plenoptic function and the elements of early vision. *Computational models of visual processing*, 1(2):3–20, 1991.

[2] S. Agarwal, K. Mierle, and Others. Ceres solver. `http://ceres-solver.org`.

[3] M. Alterman, Y. Y. Schechner, and Y. Swirski. Triangulation in random refractive distortions. In *IEEE International Conference on Computational Photography (ICCP)*, pages 1–10, Apr. 2013.

[4] O. Ben-Shahar, Y. Vasilyev, Y. Adato, and T. Zickler. Shape from specular flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32:2054–2070, 2010.

[5] C. Buehler, M. Bosse, L. McMillan, S. Gortler, and M. Cohen. Unstructured lumigraph rendering. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '01, pages 425–432. ACM, 2001.

[6] J.-X. Chai, X. Tong, S.-C. Chan, and H.-Y. Shum. Plenoptic sampling. In *Proceedings of the 27th Annual Conference*

on Computer Graphics and Interactive Techniques, SIGGRAPH '00, pages 307–318. ACM, 2000.

[7] P. Einarsson, C.-F. Chabert, A. Jones, W.-C. Ma, B. Lamond, T. Hawkins, M. Bolas, S. Sylwan, and P. Debevec. Relighting human locomotion with flowed reflectance fields. In *Proceedings of the 17th Eurographics Conference on Rendering Techniques*, EGSR '06, pages 183–194, Aire-la-Ville, Switzerland, Switzerland, 2006. Eurographics Association.

[8] T. Georgeiv, K. C. Zheng, B. Curless, D. Salesin, S. Nayar, and C. Intwala. Spatio-angular resolution tradeoffs in integral photography. In *Proceedings of the 17th Eurographics Conference on Rendering Techniques*, EGSR '06, pages 263–272, Aire-la-Ville, Switzerland, Switzerland, 2006. Eurographics Association.

[9] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen. The lumigraph. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, pages 43–54. ACM, August 1996.

[10] M. Gross and H. Pfister. *Point-Based Graphics*. Morgan Kaufmann, May 2011.

[11] S. Heber and T. Pock. Shape from light field meets robust PCA. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, *Computer Vision ECCV 2014*, number 8694 in Lecture Notes in Computer Science, pages 751–767. Springer International Publishing, Sept. 2014. DOI: 10.1007/978-3-319-10599-4_48.

[12] S. Heber and T. Pock. Convolutional networks for shape from light field. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3746–3754, June 2016.

[13] E. Iffa, G. Wetzstein, and W. Heidrich. Light field optical flow for refractive surface reconstruction. In *Proc. SPIE*, volume 8499, pages 84992H–84992H–8, 2012.

[14] M. Levoy and P. Hanrahan. Light field rendering. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, pages 31–42. ACM, 1996.

[15] K. Maeno, H. Nagahara, A. Shimada, and R.-I. Taniguchi. Light field distortion feature for transparent object recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2786–2793, 2013.

[16] P. Moulon, P. Monasse, R. Marlet, and Others. OpenMVG: An open multiple view geometry library. `https://github.com/openMVG/openMVG`.

[17] G. Nieto, F. Devernay, and J. Crowley. Variational image-based rendering with gradient constraints. In *2016 International Conference on 3D Imaging (IC3D)*, pages 1–8, Dec. 2016.

[18] J. Ponce, B. Sturmfels, and M. Trager. Congruences and concurrent lines in multi-view geometry. *Advances in Applied Mathematics*, 88:62 – 91, 2017.

[19] H. Pottmann and J. Wallner. *Computational Line Geometry*, chapter Line Congruences and Line Complexes. Mathematics and Visualization. Springer Berlin Heidelberg, 2001.

[20] A. P. Pozo, F. Briggs, and Others. Facebook surround 360. `https://facebook360.fb.com/facebook-surround-360/`.

[21] S. Pujades, F. Devernay, and B. Goldluecke. Bayesian view synthesis and image-based rendering principles. In *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3906–3913, June 2014.

[22] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, Mar. 1978.

[23] A. Sulc, A. Alperovich, N. Marniok, and B. Goldluecke. Reflection separation in light fields based on sparse coding and specular flow. In *Vision, Modelling and Visualization (VMV)*, 2016.

[24] S. Wanner and B. Goldluecke. Variational light field analysis for disparity estimation and super-resolution. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(3):606–619, Mar. 2014.

[25] G. Wetzstein, D. Roodnick, W. Heidrich, and R. Raskar. Refractive shape from light field distortion. In *2011 International Conference on Computer Vision*, pages 1180–1186, Nov. 2011.

[26] B. Wilburn, N. Joshi, V. Vaish, E.-V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy. High performance imaging using large camera arrays. In *ACM SIGGRAPH 2005 Papers*, SIGGRAPH '05, pages 765–776. ACM, 2005.

[27] B. Wilburn, V. Vaish, and Others. The (new) Stanford light field archive. `http://lightfield.stanford.edu/lfs.html`, 2017. [Online; accessed 07-March-2027].

[28] P. Zhou, L. Yu, and C. Pak. The spectrum broadening in the plenoptic function. In *Proceedings of International Conference on Internet Multimedia Computing and Service*, ICIMCS '14, pages 130:130–130:135. ACM, 2014.

[29] P. Zhou, L. Yu, and G. Zhong. The non-Lambertian reflection in plenoptic sampling. In *2013 IEEE International Conference on Image Processing*, pages 2154–2157, Sept. 2013.